

## Face Recognition Project:

**Scope:** The face detection took a major leap with deep learning techniques. We can build models with high accuracy in detecting the bounding boxes of the human face. This project will get you started with object detection and you will learn how to detect any object in an image.

Face Recognition is a technology in computer vision. In Face recognition / detection we locate and visualize the human faces in any digital image.

It is a subdomain of Object Detection, where we try to observe the instance of semantic objects. These objects are of particular class such as animals, cars, humans, etc. Face Detection technology has importance in many fields like marketing and security.

### Cascade Classifiers and Haar Features:

Cascade Classifiers and Hare Features are the methods used for Object Detection.

It is a machine learning algorithm where we train a cascade function with tons of images. These images are in two categories: positive images containing the target object and negative images not containing the target object.

There are different types of cascade classifiers according to different target objects. In our project, we will use a classifier that considers the human face to recognize it as the target object.

Haar Feature selection technique has a target to extract human face features. Haar features are like convolution kernels. These features are different permutations of black and white rectangles. In each feature calculation, we find the sum of pixels under white and black rectangles.

### Haar-cascade Detection in OpenCV:

OpenCV provides pre-trained models on Haar features and cascade classifier these models are located in OpenCV installation.

You can find the necessary XML files at:

In the below code we will see how to use these pre-trained Haar cascade models to detect Human Face. We will implement a real-time human face recognition with python.

## Open cv:

OpenCV is one of the most popular computer vision libraries. If you want to start your journey in the field of computer vision, then a thorough understanding of the concepts of OpenCV is of paramount importance. In this article, I will try to introduce the most basic and important concepts of OpenCV in an intuitive manner.

## open cv Operations:

1. Reading an image
2. Extracting the RGB values of a pixel
3. Extracting the Region of Interest (ROI)
4. Resizing the Image
5. Rotating the Image
6. Drawing a Rectangle
7. Displaying text

## Python code:

```
#importing libraries  
import cv2  
import os
```

```
#Initialize the classifier:  
cascPath=os.path.dirname(cv2.__file__)+"/data/haarcascade_frontalface_default.xml"  
faceCascade = cv2.CascadeClassifier(cascPath)
```

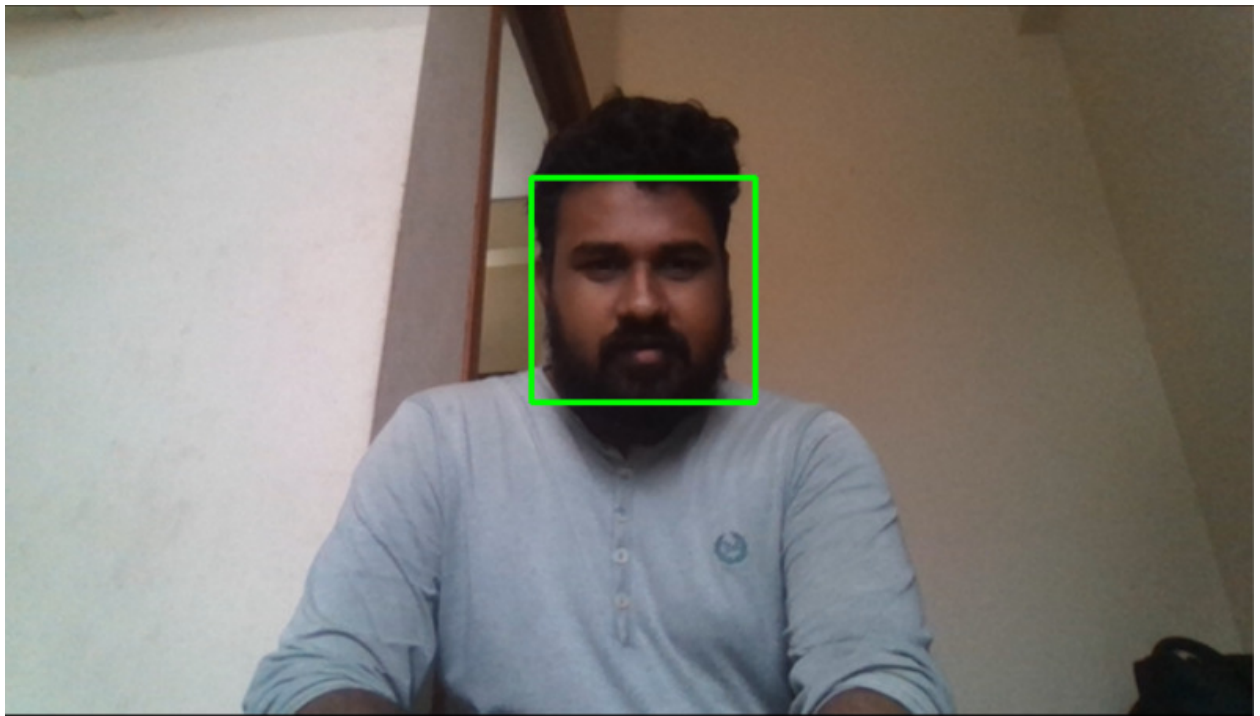
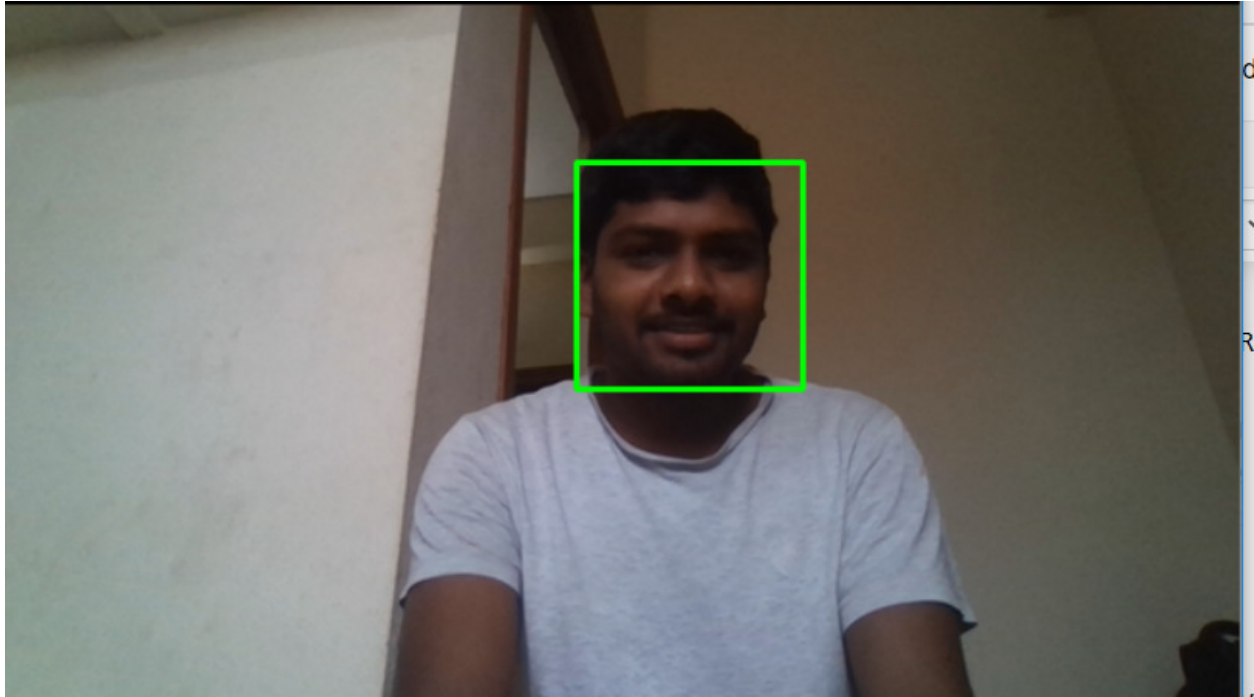
```
#Apply faceCascade on webcam frames:
video_capture = cv2.VideoCapture(0)
while True:
    # Capture frame-by-frame
    ret, frames = video_capture.read()
    gray = cv2.cvtColor(frames, cv2.COLOR_BGR2GRAY)
    faces = faceCascade.detectMultiScale(
        gray,
        scaleFactor=1.1,
        minNeighbors=5,
        minSize=(30, 30),
        flags=cv2.CASCADE_SCALE_IMAGE
    )
    # Draw a rectangle around the faces
    for (x, y, w, h) in faces:
        cv2.rectangle(frames, (x, y), (x+w, y+h), (0, 255, 0), 2)
    # Display the resulting frame
    cv2.imshow('Video', frames)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
```

---

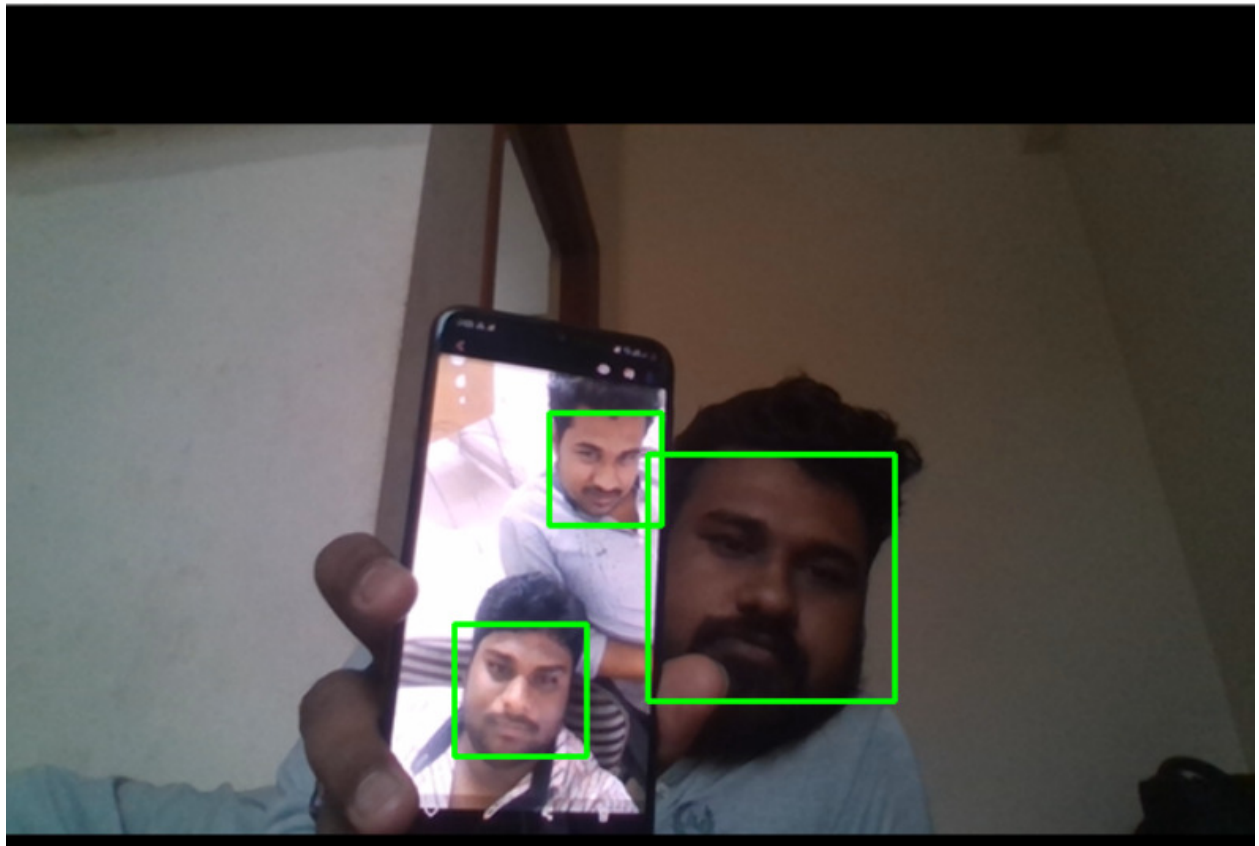
```
#Release the capture frames:
video_capture.release()
cv2.destroyAllWindows()
```

---

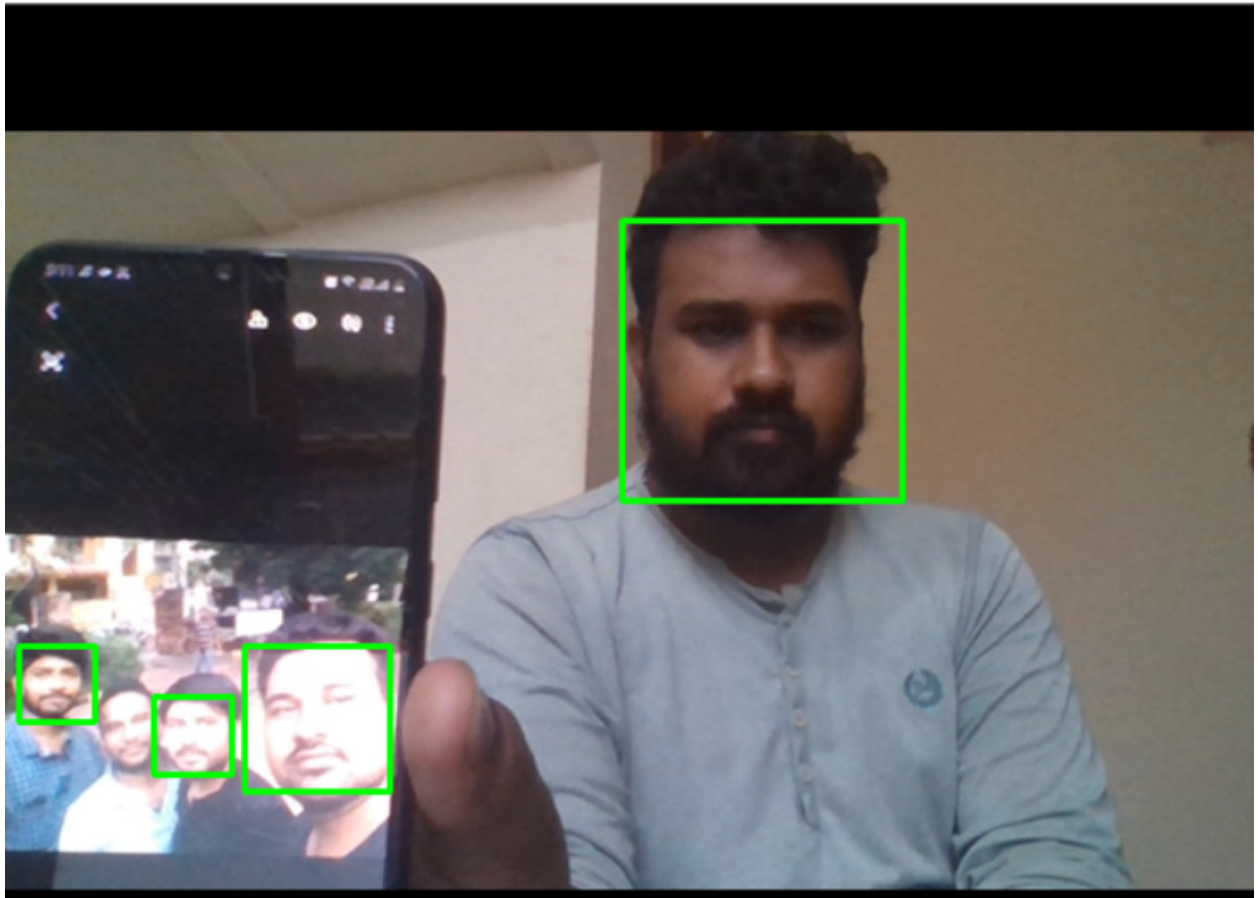
**Face detection:**



Video

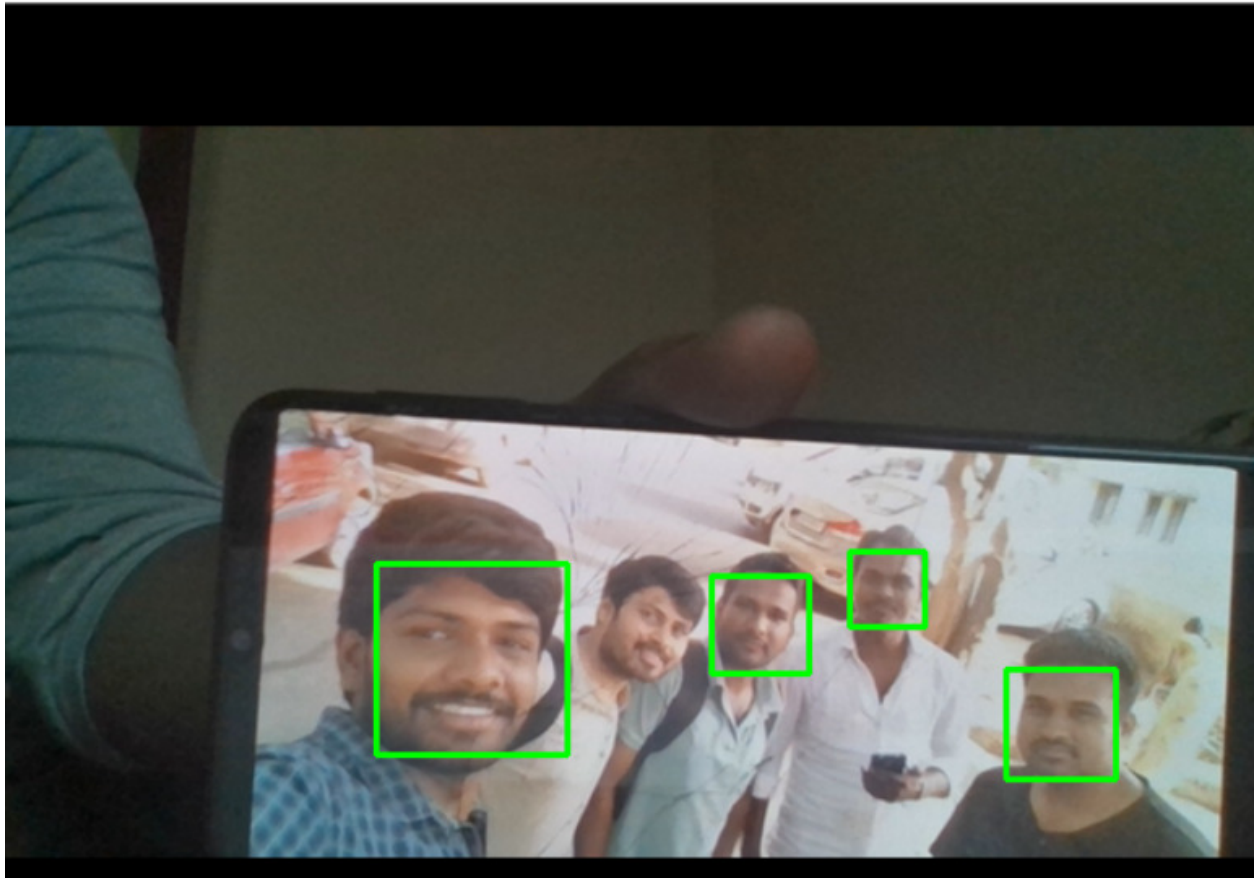


Video



Video

— □ ×



Video

